380Z Disc BASIC

RELEASE NOTE

There are three versions of BASIC supplied. These reside on one disc for an FDS system or two discs for an MDS system. The versions differ only in the provision of different levels of High Resolution Graphics.

BASICS is Extended BASIC version 5 for 380Z disc systems without support for the High Resolution Graphics option. This is supplied on disc 1 for an MDS system.

File name	BASICS.COM
BASIC version	5.0 L
Start address	100
Restart address	103
Memory limits	100 - 3260

BASICSG is Extended BASIC version 5 for 380Z disc systems with level 1 High Resolution Graphics support. This is supplied on disc 1 for an MDS system.

File name	BASICSG.COM
BASIC version	5.0 L
HRG (level 1) version	2.0 B
Start address	100
Restart address	103
Memory limits	100 - 3ED4

BASICSG2 is Extended BASIC version 5 for 380Z disc systems with level 2 High Resolution Graphics support. This is supplied on disc 2 for an MDS system.

File name	BASICSG2.COM
BASIC version	5.0 L
HRG (level 2) version	1.1 J
Start address	100
Restart address	103
Memory limits	100 - 5318

Extended BASIC version 5.0 L is an enhancement of the previous 380Z disc version 5.0 H. The differences between the two versions are given below.

1. It is no longer possible to CONTinue execution of a program following a recovery error (i.e. by jumping to location 103 from the Front Panel or from the monitor).

2. In Extended BASIC version 5.0 H a colon in a DATA statement was not parsed correctly which led to keywords after the colon being converted to their internal format. A colon no longer has this effect.

3. When GOTO or GOSUB was not the last statement on the line certain odd results could be obtained such as not transferring execution to the correct place. On rare occasions this could result in the interpreter locking up. This no longer happens.

 Copyright September 1982

4. Evaluating a Boolean expression with a string concatenation on the right hand side of the equals sign produced a 'Type mismatch' error.

   E.g. 10 IF A$ = B$ + C$ THEN ..... would not be accepted.

   This expression is now evaluated correctly.

5. Typing NEW whilst a TRACE was in operation prevented further use of the TRACE facility. This fault has been fixed in this version.

6. When renumbering an ON .. GOTO ... if one of the line numbers did not exist then the remaining line numbers in that line were not renumbered. This has now been corrected.

7. In previous versions of BASIC attempts to load a null internal format file resulted in an Out of memory error message. This fault has now been corrected.

A number of changes were made in order to produce a BASIC for use on the Network. These were also included in the 380Z disc BASIC so as to aid the transportation of programs between systems. The changes are listed below.

1. It is now possible to close input files, as well as output files. The command CLOSE INPUT #10 performs this task. When operating in a network environment this should be done when an input file has been finished with.

2. The command CLOSE #2 has been added. This has the effect of sending a CTRL-Z to the printer. Wnen operating in a network environment it is necessary to issue this command (or its equivalent) to enable the spooler to output to the printer.

The following are changes to High Resolution Graphics level 2.

1. High Resolution Graphics level 2 now supports a screen dump to the Epson MX series of printers, 80, 82 and 100, types II and III as well as the Anadex. The screen dump to both Epson and Anadex printers has been speeded up by a factor of between 2 and 6 times, depending on the pattern size used.

2. In previous versions an attempt to read a point back from the High Resolution Graphics page 1 always returned zero. This fault has now been corrected. This applies to RDOUT, COPY and DUMP.

3. It was possible to corrupt HRG workspace by dumping the whole screen or Copying large areas. This affected variables such as the current offset. This has been corrected.

4. There is a mistake in the character set shown on the last page of the supplementary manual. The first five characters on the fourth line should be < = > ? A.

 Copyright September 1982

Subroutine Initialisation

The High Resolution Graphics packages attached to BASIC now indicate which level they are and produce version numbers. They also indicate if you are running HRG BASIC in a LINK 480Z which does not have the HRG option installed. The message produced ('No HRG option installed') is also produced if any graphics CALL is attempted.

This section will be of use to those users wishing to do similar actions with your own machine code subroutines and is intended as an appendix to the manual.

The ability to give a version number results from the introduction of a new jump vector at the start of BASIC labelled INITSUB (see page 4 of this release note for its location in this version of BASIC). During BASIC's initialisation sequence it CALLs this location which is initially set up as a RET instruction. A user may patch this address to jump to his own routine, and perform any initialisation that is required, for example outputting a version number. The end of the initialisation subroutine has two possible terminations:

1) If the routines are attached to a BASIC without any other machine code subroutines then it should be terminated by a RET instruction.

2) If there are already subroutines attached to the BASIC which have patched this vector then your own initialisation routine should end by performing the code that was originally patched there.

Below is an example of the code required to patch an initialisation routine onto a BASIC without any machine code subroutines attached.

```
        ORG     INITSUB         ;The vector to patch
        JP      BEGIN           ;On initialisation jump to BEGIN

        .
        .

        ;Initialisation Code starts here
BEGIN:  LD      HL,INITMSG      ;For example
        EMT     MSG
        .
        .       ;Any other initialisation code

        RET     ;Return to BASIC
                ;If the routine is being patched to a version
                ;of BASIC which already has machine code
                ;subroutines attached then the RET instruction
                ;should be replaced by the code originally at
                ;location INITSUB.
```

This version of BASIC is one of the first with this jump vector included. Therefore care should be taken if you attempt to patch a routine which uses this vector onto an earlier version of BASIC.

The start of BASIC contains a number of jumps and vectors. These will be of interest to those who wish to add machine code subroutines to the interpreter. This listing contains the values of all the symbolic locations mentioned in chapter 13. The addresses of the routines themselves will vary from version to version, and are here represented as zeros.

```
0100 C30000   0165 BASIC:   JP      INIT      ;INITIALIZE ENTRY POINT
0103 C30000   0166 REST:    JP      ERR.RC    ;RECOVERY ENTRY POINT
0106 C30000   0167 USR:     JP      ERR.FC    ;USER DEFINED JUMP
0109 C30000   0168 USRVF:   JP      CVRLIN    ;USR SUBROUTINE TO GET VAR
010C C30000   0169 USRVT:   JP      CS16R2    ;USR SUBROUTINE TO PASS VAR
010F C30000   0170 SAVVAR:  JP      SAVINT    ;SAVE INTEGER IN VAR
0112 C30000   0171 OUTMX:   JP      .OUTM     ;PRINT MESSAGE
0115 CD0000   0172 E.ERR:   CALL    ER.EXT    ;EXTERNAL ERROR
0118 0000     0173 BBUFV:   DEFW    ENDIT     ;END OF BASIC
011A 0000     0174 SUBPTR:  DEFW    0         ;POINTER TO SUBS
011C 0000     0175 ENDMEM:  DEFW    0         ;END OF MEMORY
011E 0000     0176 BRFCB:   DEFW    RFCB      ;READ FCB
0120 0000     0177 BWFCB:   DEFW    WFCB      ;WRITE FCB
0122 424153   0178 DEFEXT:  DEFM    'BAS'     ;DEFAULT EXTENSION
0125 C9       0179 INITSUB:RET                ;USER INITIALISATION VECTOR
0126 00       0180         NOP
0127 00       0181         NOP
```